UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/004,197	11/14/2001	Craig Nemecek	CYPR-CD01220M	1794

7590 11/23/2005

WAGNER, MURABITO & HAO LLP
Third Floor
Two North Market Street
San Jose, CA 95113

EXAMINER	
PROCTOR, JASON SCOTT	

ART UNIT	PAPER NUMBER
2123	

DATE MAILED: 11/23/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/004,197	Applicant(s) NEMECEK ET AL.	
	Examiner Jason Proctor	Art Unit 2123	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 September 2005*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-21* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-21* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *14 November 2001* is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

Claims 1-21 were rejected in Office Action dated 6 June 2005. Applicants have submitted a response dated 6 September 2005, in which claims 1, 4, and 5 were amended. Claims 1-21 are submitted for reconsideration.

Claims 1-21 are rejected.

Claim Objections

The previous objection to claim 1 has been overcome by Applicants' amendment. The previous objection has been withdrawn.

Claim Rejections - 35 USC § 112

The previous rejections of claims 3-6, 16-19 and 21 under 35 U.S.C. § 112, first paragraph, as failing to comply with the written description requirement are withdrawn in light of Applicants' response. The Examiner thanks Applicants for carefully identifying the support for these limitations in the disclosure, especially in light of the Request for Non-Publication filed in this application.

The previous rejections of claims 3-6 under 35 U.S.C. § 112, second paragraph, are withdrawn in light of Applicants' response.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. § 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

1. Claims 1-21 are rejected under 35 U.S.C. § 103(a) as being unpatentable over US Patent No. 5,911,059 to Profit, Jr. (Profit) in view of "DEBUG" as described in "A DEBUG Tutorial" by Daniel B. Sedory, copyright 2004 (Sedory) and further in view of "Microsoft PressPass – Microsoft Files Summary Judgement Motions" by Microsoft® published February 12, 1999 (Microsoft).

Regarding claim 1, Profit teaches an in-circuit emulation system comprising:

a processor having a microcontroller clock (Fig. 7, reference 204; column 6, lines 5-24; regarding a clock in the target program in the processor, column 12, lines 24-28);

a virtual processor (referred to as a processor model shell 212) (column 6, lines 25-48) operating in lock step synchronization with the processor (column 11, lines 40-43);

a gatekeeper circuit (referred to as RUN/HALT controller 240) coupled to the virtual processor and the processor (Fig. 8, reference 240; column 8, line 65 – column 10, line 31); and

a host computer running in-circuit emulation debug software (Fig. 7, reference 214; column 6, lines 49-60) in communication with the gatekeeper circuit so that halt commands are passed through and regulated by the gatekeeper circuit (Fig. 7, reference 222; column 9, lines 4-6).

Official notice is taken that the term *microcontroller* refers to a single unit usually comprising central processing unit, memory, and I/O ports. As Profit teaches an emulator unit that contains at least these features (Fig. 7, reference 202), it would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention that Profit's emulator is readily adaptable to accept microcontrollers, as would be desired by a person whose goal it is to develop and debug code for microcontrollers.

Profit does not explicitly recite that requests for data from the virtual processor (to the actual processor) are passed through the gatekeeper circuit, however Profit does explicitly teach the use of standard software debugging tools (column 6, lines 49-60).

The Sedory reference describes the DEBUG command of Microsoft® MS-DOS® operating system versions 5.0 and later (Sedory, page 1). The Microsoft reference establishes the release date of MS-DOS® 5.0 as June 1991 (Microsoft, page 4). Therefore the Sedory reference is relied upon as describing the DEBUG command of MS-DOS® 5.0 as it was known in the art in June 1991.

Sedory teaches that the DEBUG command includes the capability of viewing memory contents (page 3, Dump command), modifying memory contents (pages 4-5, Enter command; page 8, Move command), viewing registers (pages 9-10, Register command), and modifying registers (pages 9-10, Register command). Additionally, these concepts are well known in the art of debugging as standard techniques, commonly referred to as traces, tracing, memory dumping, writing to memory locations, et cetera.

It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention in combination with his own knowledge of the particular art, at the explicit suggestion of Profit to combine software debugging tools, to incorporate the well known debugging techniques embodied in DEBUG with the in-circuit emulation system taught by Profit. It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention to make the requisite modifications to the features of DEBUG to accommodate the communicative nature of Profit's in-circuit emulation system. This combination would render obvious the recited limitations of communicating "halt commands and requests for data from the virtual computer" through the gatekeeper circuit taught by Profit.

Applicants have not seasonably traversed this use of Official Notice therefore it is regarded as admitted prior art according to MPEP 2144.03(c).

In response, Applicants' argue primarily that:

> Applicant contends that Profit fails to disclose a virtual microcontroller running in lock-step synchronization with the microcontroller, as claimed. Applicant further contends that Profit fails to disclose a host computer running In-Circuit Emulation debug software, as claimed.
> [...]
> From the example provided, the hardware simulator of Profit is principally a processor model shell, which simulates activity at the target processor's pins; however, it does not emulate the processor's functionality, see col. 6, ln. 25-48. The hardware simulator of Profit is not a virtual microcontroller.
> [...]
> [T]he portion of Profit referenced to demonstrate In-Circuit Emulation debug software does not disclose the operation of In-Circuit Emulation debug software, but rather states a generalization about what software might be running on the host computer, and provides an example of a software package for designing the target *circuitry*; see col. 6, ln. 49-60.

The Examiner respectfully traverses this argument as follows.

To address Applicants' first argument, Applicants' attention is respectfully drawn to Profit, column 6, lines 27-32, emphasis added:

> The hardware simulator 210 is a conventional software program that simulates the electrical and logical activity of the target circuitry as seen by the target processor *[target processor 204 (see column 6, lines 19-24)]*. The hardware simulator 210 includes a **processor model shell 212 which converts a sequence of processor functions** to activity at simulated pins of the target processor [*target processor 204*]. Such a **sequence of processor functions corresponds to an instruction executed in the target program 22** [...]

Applicants' attention is respectfully drawn to Profit, column 6, lines 12-15, emphasis added:

> **The processor 204** communicates with the memory 206 **to receive and execute computer instructions, including those in the target program 22**, and to write data and read data from the memory 206.

The Examiner finds no evidence that the processor model shell "does not emulate the processor's functionality." Profit explicitly discloses that the processor model shell converts a sequence of processor functions, which correspond to an instruction executed in the target program by processor 204, into activity at the simulated pins of the target processor. The hardware simulator of Profit clearly contains a virtual microcontroller, specifically processor model shell 212.

To address Applicants' second argument, Applicants' attention is respectfully drawn to Profit, column 5, lines 51-57:

> The system of the present invention also allows the extensive use of existing debugging tools to aid the developer in the development and integration of the target system. The system combines interacting elements of hardware and executing software, in part by physical emulation and in part by abstract software simulation.

The Examiner finds no evidence that Profit fails to disclose "In-Circuit Emulation debug software". Profit clearly discloses an in-circuit emulation system [*interacting elements of hardware and executing software, in part by physical emulation and in part by abstract software simulation*] to be used with existing debugging tools. Clearly Profit is disclosing the use of existing debugging tools to be used with the disclosed in-circuit emulation system. Applicants' have neither disclosed nor claimed a particular invention specifically related to "In-Circuit Emulation debugging software," and the Examiner can find no credible distinction between Profit's disclosure of debugging tools and Applicants' claimed "In-Circuit Emulation debugging software."

Applicants' arguments have been fully considered but have been found unpersuasive.

Regarding claim 2, Profit teaches a gatekeeper clock running independent of the microcontroller clock to clock operations carried out in the gatekeeper circuit (column 10, lines 38-41, "In this embodiment, the simulation time keeper circuit 232 includes a counter"; column 10, lines 44-45, "The counter is driven by the clock signal on line 242").

Regarding claim 3, Profit teaches that the gatekeeper circuit comprises means for halting the microcontroller, functionally equivalent to placing it in a "sleep state" (column 8, line 65 – column 10, line 31; column 10, line 32 – column 11, line 7; especially column 9, lines 41-46). When placing a microcontroller into a sleep state, the gatekeeper circuit is reasonably apprised that the microcontroller is in a sleep state, obviating the purpose of a specialized sleep state detection ability.

Regarding claims 4 and 5, Profit teaches that the gatekeeper circuit (RUN/HALT controller 240) monitors the state of the microcontroller (column 9, lines 47-55). In another embodiment, Profit teaches that the target bus watch circuit 224 (which comprises, among other components, the RUN/HALT controller 240) monitors "the data address and status lines on the target bus 208 of the processor emulator 202" (column 10, lines 4-6). It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention, in combination with his own knowledge of the particular art as well as Profit's explicit teaching of the advantages of various embodiments, to combine and modify the teachings of Profit to arrive at the claimed invention.

Regarding claim 6, Profit teaches that the gatekeeper circuit notifies the host computer of the status of the microcontroller (column 10, lines 4-23).

Regarding claim 7, Profit teaches sending a halt command (referred to as HOST INTERRUPT request) to the microcontroller (referred to as processor emulator 202 which includes processor 204 of Fig. 7) that is received from the virtual microcontroller (referred to as hardware simulator 210 which includes processor model shell 212 of Fig. 7) and halting the microcontroller or virtual microcontroller in response to the halt command (column 8, lines 13-16; column 9, lines 40-61; column 10, lines 11-18).

Regarding claim 8, Profit teaches that the gatekeeper circuit (RUN/HALT controller 240) monitors the state of the microcontroller (column 9, lines 47-55). In another embodiment, Profit teaches that the target bus watch circuit 224 (which comprises, among other components, the RUN/HALT controller 240) monitors "the data address and status lines on the target bus 208 of the processor emulator 202" (column 10, lines 4-6). Profit teaches that one of the tasks of the RUN/HALT controller 240 is to halt the microcontroller and "communicate required event information between the processor emulator 202 and the hardware simulator 210" (column 10, lines 4-18). Profit therefore teaches detecting that a halt has occurred in the microcontroller and notifying the host computer that a break has occurred.

Regarding claims 9 and 10, the recited limitations are equivalent to a break (as defined by Microsoft Computer Dictionary, Fifth Edition) in claim 9 and a breakpoint or breakpoint instruction (as defined by The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition) in claim 10. Profit explicitly teaches the inclusion of standard software debugging tools (column 6, lines 49-60) which a person of ordinary skill in the art at the time of Applicants' invention would recognize as including both a break as well as a breakpoint or breakpoint instruction. It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention, at the explicit suggestion of Profit, to include well-known and basic concept of debugging in the in-circuit emulation system taught by Profit. Regarding the recitation of a "breakpoint controller" (claim 10), Profit teaches a RUN/HALT controller 240 that is functionally equivalent (column 10, lines 4-23).

Regarding claim 11, the recited limitations are equivalent to a break (as defined by Microsoft Computer Dictionary, Fifth Edition) in claim 9 and a breakpoint or breakpoint instruction (as defined by The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition) in claim 10. Profit explicitly teaches the inclusion of standard software debugging tools (column 6, lines 49-60) which a person of ordinary skill in the art at the time of Applicants' invention would recognize as including both a break as well as a breakpoint or breakpoint instruction. It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention, at the explicit suggestion of Profit, to include well-known and basic concept of debugging in the in-circuit emulation system taught by Profit.

Further, the combination formed in the rejection of claim 1 teaches the limitations of "permitting access to registers and memory locations". Specifically, Sedory teaches that the DEBUG command includes the capability of viewing memory contents (page 3, Dump command), modifying memory contents (pages 4-5, Enter command; page 8, Move command), viewing registers (pages 9-10, Register command), and modifying registers (pages 9-10, Register command).

Regarding claim 12, the recited limitations are equivalent to a break (as defined by Microsoft Computer Dictionary, Fifth Edition). Profit explicitly teaches the inclusion of standard software debugging tools (column 6, lines 49-60), which a person of ordinary skill in the art at the time of Applicants' invention would recognize as including a break. It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention, at the explicit suggestion of Profit, to include well-known and basic concept of debugging in the in-circuit emulation system taught by Profit.

Claim 13 recites the method employed by the system of claims 1. To that effect, Profit teaches an in-circuit emulation system and accompanying method comprising:

a virtual processor (referred to as a processor model shell 212) (column 6, lines 25-48) operating in lock step synchronization with a processor (column 11, lines 40-43);

a gatekeeper circuit (referred to as RUN/HALT controller 240) coupled to the virtual processor and the processor (Fig. 8, reference 240; column 8, line 65 – column 10, line 31); and

a host computer running in-circuit emulation debug software (Fig. 7, reference 214; column 6, lines 49-60) in communication with the gatekeeper circuit so that halt commands are passed through and regulated by the gatekeeper circuit (Fig. 7, reference 222; column 9, lines 4-6).

Official notice is taken that the term microcontroller refers to a single unit usually comprising central processing unit, memory, and I/O ports. As Profit teaches an emulator unit that contains at least these features (Fig. 7, reference 202), it would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention that Profit's emulator is readily adaptable to accept microcontrollers, as would be desired by a person whose goal it is to develop and debug code for microcontrollers.

Profit does not explicitly recite that requests for data from the virtual processor (to the actual processor) are passed through the gatekeeper circuit, however Profit does explicitly teach the use of standard software debugging tools (column 6, lines 49-60).

The Sedory reference describes the DEBUG command of Microsoft® MS-DOS® operating system versions 5.0 and later (Sedory, page 1). The Microsoft reference establishes the release date of MS-DOS® 5.0 as June 1991 (Microsoft, page 4). Therefore the Sedory reference is relied upon as describing the DEBUG command of MS-DOS® 5.0 as it was known in the art in June 1991.

Sedory teaches that the DEBUG command includes the capability of viewing memory contents (page 3, Dump command), modifying memory contents (pages 4-5, Enter command; page 8, Move command), viewing registers (pages 9-10, Register command), and modifying registers (pages 9-10, Register command). Additionally, these concepts are well known in the art of debugging as standard techniques, commonly referred to as traces, tracing, memory dumping, writing to memory locations, et cetera.

It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention in combination with his own knowledge of the particular art, at the explicit suggestion of Profit to combine software debugging tools, to incorporate the well known debugging techniques embodied in DEBUG with the in-circuit emulation system taught by Profit. It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention to make the requisite modifications to the features of DEBUG to accommodate the communicative nature of Profit's in-circuit emulation system. This combination would render obvious the recited limitations of communicating "halt commands and requests for data from the virtual computer" through the gatekeeper circuit taught by Profit.

Applicants have not seasonably traversed this use of Official Notice therefore it is regarded as admitted prior art according to MPEP 2144.03(c).

In response, Applicants' reiterate the arguments submitted regarding claim 1, which have been addressed above. These arguments have been fully considered but have been found unpersuasive.

Regarding claims 14 and 15, the recited limitations are equivalent to a break (as defined by Microsoft Computer Dictionary, Fifth Edition) in claim 9 and a breakpoint or breakpoint instruction (as defined by The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition) in claim 10. Profit explicitly teaches the inclusion of standard software debugging tools (column 6, lines 49-60) which a person of ordinary skill in the art at the time of Applicants' invention would recognize as including both a break as well as a breakpoint or breakpoint instruction. It would have been obvious to a person of ordinary skill in the art at the time of

Applicants' invention, at the explicit suggestion of Profit, to include well-known and basic concept of debugging in the in-circuit emulation system taught by Profit. Regarding the recitation of a "breakpoint controller" (claim 10), Profit teaches a RUN/HALT controller 240 that is functionally equivalent (column 10, lines 4-23).

Regarding claim 16, Profit teaches that the gatekeeper circuit comprises means for halting the microcontroller, functionally equivalent to placing it in a "sleep state" (column 8, line 65 – column 10, line 31; column 10, line 32 – column 11, line 7; especially column 9, lines 41-46). When placing a microcontroller into a sleep state, the gatekeeper circuit is reasonably apprised that the microcontroller is in a sleep state, obviating the purpose of a specialized sleep state detection ability.

Regarding claims 17 and 18, Profit teaches that the gatekeeper circuit (RUN/HALT controller 240) monitors the state of the microcontroller (column 9, lines 47-55). In another embodiment, Profit teaches that the target bus watch circuit 224 (which comprises, among other components, the RUN/HALT controller 240) monitors "the data address and status lines on the target bus 208 of the processor emulator 202" (column 10, lines 4-6). It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention, in combination with his own knowledge of the particular art as well as Profit's explicit teaching of the advantages of various embodiments, to combine and modify the teachings of Profit to arrive at the claimed invention.

Regarding claims 19 and 20, Profit teaches that the gatekeeper circuit (RUN/HALT controller 240) monitors the state of the microcontroller (column 9, lines 47-55). In another embodiment, Profit teaches that the target bus watch circuit 224 (which comprises, among other components, the RUN/HALT controller 240) monitors "the data address and status lines on the target bus 208 of the processor emulator 202" (column 10, lines 4-6). Profit teaches that one of the tasks of the RUN/HALT controller 240 is to halt the microcontroller and "communicate required event information between the processor emulator 202 and the hardware simulator 210" (column 10, lines 4-18). Profit therefore teaches notifying the host computer of the state of the microcontroller and virtual microcontroller, whether that state is halted, sleep, or otherwise.

Claim 21 recites the method employed by a system combining the limitations of claims 5, 6, 8, 9, 10, and 11. As Profit in view of DEBUG renders all of these limitations obvious, Profit in view of DEBUG similarly renders the combination of these limitations obvious. Profit in view of DEBUG teaches the system and its operation, thereby rendering the method of its use obvious. It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention to combine the teachings of Profit in view of DEBUG in the combination recited by claim 21 in order to achieve the best features of the prior art. Motivation to do so would be found in the knowledge of a person of ordinary skill in the art.

In response, Applicants' reiterate the arguments submitted regarding claim 1, which have been addressed above. These arguments have been fully considered but have been found unpersuasive.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Art considered pertinent by the examiner but not applied has been cited on form PTO-892.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason Proctor whose telephone number is (571) 272-3713. The examiner can normally be reached on 8:30 am-4:30 pm M-F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Leo Picard can be reached at (571) 272-3749. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Jason Proctor
Examiner
Art Unit 2123

jsp

Paul L. Rodriguez 11/18/05
Primary Examiner
Art Unit 2125